Phase-Accumulator based Multi-Channel High-Precision Digital PWM Architecture

Hermann Meuth, Ireneusz Janiszewski
Electr. Engineering and Information Technology
University of Applied Sciences
Darmstadt, Germany
Email: {meuth, janiszewski}@fh-darmstadt.de

Kai Schade
Research and Technology, Facility 050
DaimlerChrysler AG
Böblingen, Germany
Email: kai.schade@daimlerchrysler.com

***Abstract*— A novel fully digital accumulator based pulse-width modulation (PWM) architecture operates in frequency domain, permitting to autonomously control frequency and phase parameters independently without the need for additional processor intelligence, e.g. in real-time time-critical applications. The fully digital design is available in text-based hardware design language (HDL), offering flexibility in technology implementation. High-precision sample implementations include 0.35μm CMOS ASIC, CPLD, and FPGA. For clock rates in excess of 100 MHz, pulse step widths of 10 ns and a digital settability of sub-Hertz and fractions of degrees in frequency and phase resolution are realistic. The architecture allows for cascading an in principle unlimited number of synchronous channels rigid in frequency and phase, subject only to available chip or logic resources. Finally, implemented as ASIC, highest clock rates are conceivable either by quartz, or by an on-chip ring oscillator, with the PWM carrier tuned digitally to an external (lower frequency) reference.**

I. Introduction

Pulse-width modulated (PWM) signal generators periodically deliver, traditionally at constant frequency, pulse trains of constant level, yet variable pulse width (or pulse length in time). Pulse-width generators thus generate signals of fixed voltage, periodically turned on and off. The pulse width may be varied in duty cycle, i.e. the ratio of on-time to off-time, or on-time to period. Instead of an 'on-off' sequence, switching between two voltage levels is also conceivable, but less common.

Whereas regular signal generation produces *amplitude-value* sequences in time, PWM generates *pulse-length* sequences. Here as there, these sequences are used to carry the actual signal information. However, to achieve signal quality, regular signal generation imposes high demands on circuit and transmission integrity, and usually on linearity and reproducibility. PWM, in contrast, delivers only two different signal levels, one of which usually being zero, and is thus directly suitable to be dealt with by binary fully digital schemes, including embedded implementations [1], essentially without analog circuitry. Due to the binary on-off characteristics, PWM is rather immune to noise in signal level, except timing jitter. On the other hand, PWM involves regularly much higher (carrier) frequencies, and even higher signal bandwidth due to demands on rise times, than would be expected from the application's natural range of frequencies. PWM encoding is thus limited only by the digital circuitry's bandwidth. Moreover, PWM requires high *timing* (clocking) resolution to achieve precision. When followed by an integrating element, as e.g. a simple capacitive load, the PWM pulse sequence in time may be converted to a signal amplitude sequence in time, offering simple digital-to-analog conversion with minimum analog complexity all the way to involved signal-processing schemes [2], [3]. PWM therefore provides a powerful, flexible yet straight-forward technique to drive analog circuits with digital signals, which is cost effective and readily implemented in a broad range of high-clock rate digital technologies, as modern field-programmable logic devices.

This work was supported by the German Ministry of Research and Technology under contract FKZ17 103 03

II. Representative Applications

These features have led to many practical applications of PWM today. This fact is essentially driven by semiconductor power switching technology. For power generation for analog, i.e. continuously varying signals, a large fraction of power has to be picked up by the amplifier's transistor, operated in class A, AB or B, whereas on-off switching places only moderate power demands on the switching device, then operated in so-called class D mode. The class D technique increasingly finds its way into audio applications, from providing high-end high-power sound at large open-air events all the way to inexpensive battery-powered consumer gadgets as MP3 players. Other applications range from AC-DC and AC-AC (switched) power converters, where, again, a certain voltage level is produced by PWM 'chopping' with subsequent capacitive averaging, and electro-mechanical actuator drives and controls, and a host of power drive schemes [4], [5], specifically electric motor drives [6]. Employing PWM, power factor control (PFC) [7] reduces undesirable reactive feedback into the power grid.

A number of novel-demand applications as electrical vehicle automotive drives are actively being explored [8], [9]. While simple applications call only for the variation of the PWM duty cycle, more involved schemes continue to set new demands on signal quality and resolution, as well as on interfacing and configurability. More recently, there have emerged needs also for modulating both frequency and phase, including harmonic

0-7803-9052-0/05/$20.00 © 2005 IEEE.

[10] and random [11], [12], and even more so for multi-channel capability [13].

PWM has thus become a regular control building block and therefore a common peripheral component for microprocessors and digital signal processors, with continuously growing functional features, as modulation schemes, to open the door to many new industrial and consumer applications.

III. Accumulator-Based Architecture

Traditionally, pulse-width modulators were based on analog circuitry, comprising, e.g., a comparator, which is fed by a reference threshold value into its one input, and with a linear voltage sweep into the other, the latter constituting a (saw-tooth-shaped) carrier waveform. With the advent of dedicated digital signal processor cores, and, more generally, of digital signal processing techniques, fully digital schemes have taken over, except for highest frequencies not (yet) reachable by current digital circuitry. Such fully digital PWM schemes (DPWM) offer many advantages in precise settability, reproducibility, lack of drifting and ease of digital interfacing and control, while a price is to be paid for high timing precision and thus high frequency clock rates, and the requirement for minimum slew of the pulses' rising and falling edges.

Current traditional architectures regularly contain at their core counter-based timers, basically generating saw tooth sequences, setting a timing grid via counter reset upon end-point, at which the pulses may be turned on or off, thus operating effectively in time domain. Timing markers on this grid usually have to be managed with event managers, for which a real-time support from a processor is indispensable. Pulse width modulators available on the market thus generally feature integrated processors, as, e.g. Intersil's HIP63xx or ISL65xx, or DSPs, as e.g. Texas Instruments' TMS320x280x, with on-chip specialized PWM units and an 'Event Manager' to produce space vectorized signals [14], [15].

The architecture presented here utilizes an accumulator in lieu of the counter, [16]. By accumulator, cf. Fig. 1, an algorithmic summing block is meant, whose sum output is fed back to one of the two summand inputs. This in turn produces

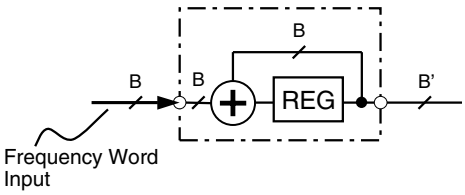

Fig. 1. The Phase-accumulator.

a saw tooth with variable slope, digitally controlled by the second summand input, see Fig. 2. The frequency of periodic time base is thus generated by the summand input, and not by a reset value, thus operating effectively in frequency domain.

In Table I, the various features of an accumulator based design and of a counter based design are juxtaposed. The arrows in middle column of the table indicates better solution.

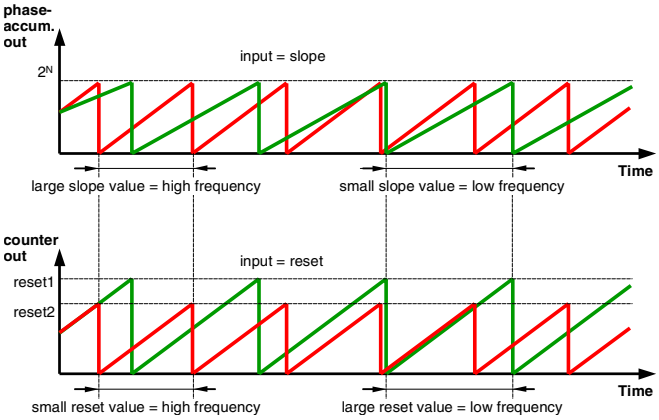

Fig. 2. Phase-accumulator versus counter based architecture waveforms.

TABLE I
Features of Phase-Accumulator Versus Counter Based PWM

Phase-Accumulator		Counter
PWM is indirectly generated from saw-tooth waveform	⇔	PWM is indirectly generated from saw-tooth waveform
Periodicity is derived from reaching overflow over 2N	⇐	Periodicity is derived from reaching reset value
Multi-bit increments of accumulator	⇒	Counter increments by 1
Simple design	⇐	Moderately complex design
Basic building block: wide-bit-input adder and register	⇐	Basic building block: wide-bit-input incrementer, comparator and register
Input: saw-tooth slope value	⇔	Input: counter reset value
Input vs. signal frequency: linear	⇐	Input vs. signal frequency: hyperbolic
Frequency resolution independent of frequency (1 LSB)	⇐	Frequency resolution depends of frequency
Frequency and phase independent	⇐	Frequency and phase dependent
Frequency domain scheme	⇔	Time domain scheme

Comparing specifically setting the phase values in both counter and accumulator schemes, it is apparent, that, for the counter scheme, changing the frequency value via the reset point, this implies also a change in phase, which thus has to be tracked in order to remain correct. With the accumulator scheme, the natural independency of frequency and phase is automatically maintained, as a full 2π–swing is realized by the accumulator's overflow, say at 2^B for an input of B bits, and an intermediate phase value $0 < \varphi < 2\pi$ retains its value irrespective of any frequency, i.e. slope value change. Fig. 3 shows, how the saw tooth sequence is transformed into a pulse-width sequence. In-flight changes of PWM frequency, pulse width or pulse phasing are shown, and are easily realized within the presented scheme. Simply speaking, the pulse is turned on, when a given phase 'SET' value is exceeded, and turned again off, when a given 'RESET' value is exceeded. It is clear, that a certain phase value may not be hit directly, as the summing accumulator could jump over it. Hardware

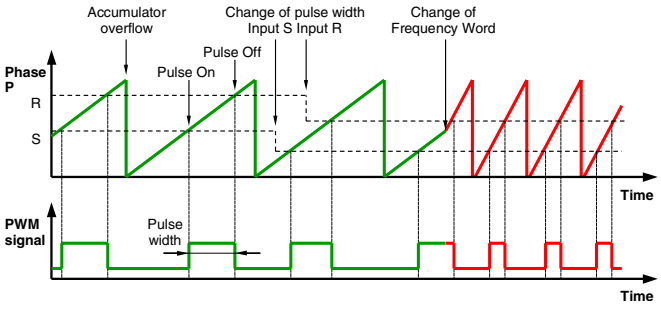

Fig. 3. Saw-tooth to pulse-width.

wise, comparators are thus used for controlling pulse 'ON' and 'OFF' times, whose second inputs are fed with the corresponding phase values, see Fig. 4.

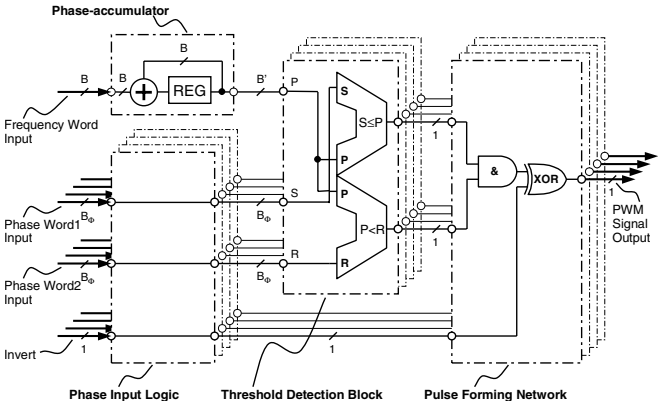

Fig. 4. PWM generator with external 'invert' (see Section V) control signal.

With this scheme, independent phase modulation and frequency modulation are thus feasible simply by entering the wanted frequency and phase value sequences.

IV. Numerical Design Parameters

The design being fully digital, its performance may be precalculated, and exact formulae may be given to determine its signal characteristics. Specifically, the PWM-signal frequency f_{PWM} is given by the clock frequency f_c in Hertz, the dimensionless binary frequency input word of bit width B, $F^b = \{F_{B-1}, \ldots, F_1, F_0\}$, corresponding to a dimensionless decimal frequency word F^d to yield

$$f_{PWM} = F^d \frac{f_c}{2^B}, \text{ with } F^d = \sum_{i=0}^{i=B-1} F_i 2^i, \quad (1)$$

where $F_i \in \{0,1\} \forall i = 0 \ldots (B-1)$.

The smallest settable frequency increment or decrement thus results for $F^d = 1$, i.e. $F_0 = 1$, all other $F_i = 0$, $\forall i = 1 \ldots (B-1)$, as

$$\Delta f = \frac{f_c}{2^B} \quad (2)$$

The time grid of the PWM signal is principally determined by the clock rate. For every pulse level transition, we have $\Delta t = 1/f_c$, and thus for the pulse ON-time $2 \times \Delta t = 2/f_c$.

The principal phase accuracy on the basis of this timing grid may as well be calculated to

$$\Delta\varphi_F = \frac{f_{PWM}}{f_c} \times 360 = \frac{F^d}{2^B} \times 360 \quad \text{in degrees, or} \quad (3)$$

$$\Delta\varphi_F = \frac{f_{PWM}}{f_c} \times 2\pi = \frac{F^d}{2^B} \times 2\pi \quad \text{in rad.} \quad (4)$$

The accumulator's output value, cf. Fig. 4, may be passed on to the phase comparator input in full bit width B, or LSB-truncated to a suitable width $B' < B$. In this case, the phase accuracy limit will change to lower accuracy, where the exponent B is to be replaced by the value B' in the above equation.

The resulting combined phase error $\Delta\varphi \leq \Delta\varphi_F + \Delta\varphi_\Phi$ is composed of $\Delta\varphi_F$ and of the phase threshold settability $\Delta\varphi_\Phi$ of the phase word inputs 1 and 2, Fig. 4, of bit width $B_\Phi \leq B$, where

$$\Delta\varphi_\Phi = \frac{1}{2^{B_\Phi}} \times 360 \quad \text{in degrees, or} \quad (5)$$

$$\Delta\varphi_\Phi = \frac{1}{2^{B_\Phi}} \times 2\pi \quad \text{in rad.} \quad (6)$$

These formulae show, that, depending only on the clock frequency f_c and the various bit-widths B, B', or B_Φ PWM performance is at the hands of the designer.

V. Design Details and Implementations

The overall scheme, see Fig. 5, or Fig 4, permits considerable flexibility in adapting to various needs regarding interfacing and I/O formats. The scheme is composed of various building block, i.e. the phase accumulator block, the phase input logic, the threshold detection block, and the pulse forming network. The block scheme offers considerable flexibility to the designer. Each block may be detailed, depending on the specific application. Unless input data are checked by an external software check, the pulse-ON point in time may trail the pulse-OFF point in time, leading to inverted pulses, which must be corrected. Frequently, when a PWM is used in a fully hardware based circuit, such software based intercept is not possible, and the invert correction must be executed by hardware, either via an invert bit derived from other parts of the circuit, or by the PWM circuit itself, i.e. autonomously. For this end, in addition to the aforementioned blocks, the circuit may need an additional control input for the invert function (invert bit).

In Fig. 5, the multi-bit channel P represents the actual phase, tracking the mentioned saw tooth pattern of Fig. 3. On the other hand, the multi-bit channel S represents the phase value, at which the pulse is to be turned on, while being turned off at the phase value R.

In the following we specify five sample implementations, where in part, their differences may be only subtle, while offering different options to the designer or user.

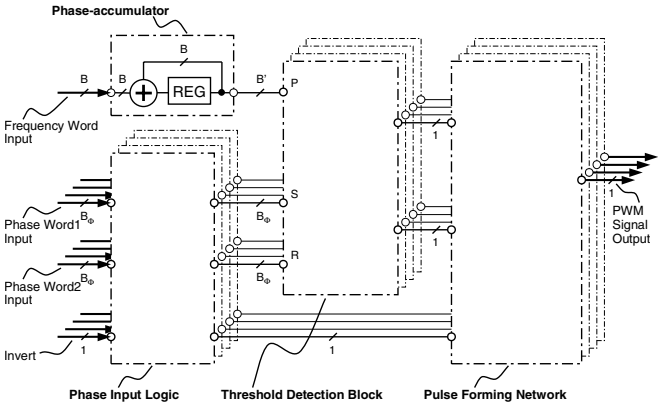

Fig. 5. PWM generator on purely digital base - overall scheme.

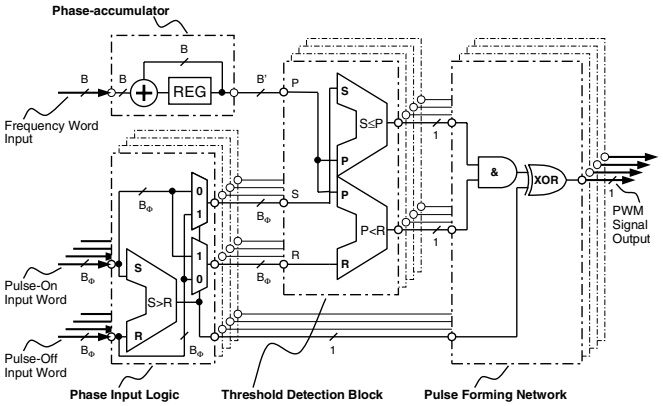

Fig. 6. PWM generator, with phase-ON and OFF inputs, without invert control signal.

A. Software-based signal control with state-controlled pulse forming network

This implementation was already shown in Fig. 4. Here, it must be ascertained by external checking routine for the phase input data, that no ON-OFF overpass takes place. The threshold detection block contains two comparators, one switching high for $P \geq S$, and the other for $P < R$. The pulse is high, as long as both comparators are high via an AND gate. Finally, the subsequent EXOR constitutes a 'switchable' inverter, inverting, when the invert bit is high. As can be seen, the pulse forming network is asynchronous. This may be an issue, when PWM signals latched to an external clock may be needed.

B. Phase-ON and Phase-OFF inputs, without invert bit and with state-controlled pulse forming network

In this implementation, see Fig. 6, the input values for phase ON and phase OFF are first checked by the phase input logic block whether ON trails OFF, in which case an invert bit is generated autonomously. For this end, the phase input values for pulse ON and OFF are first entered into a comparator, and then, if ON trails OFF, interchanged via two multiplexers, to be suitable for the following comparator logic. To yield the correct pulse for ON trailing OFF, the pulse forming must include an inversion. The remaining blocks are implemented the same as in Section V-A.

C. Phase-ON and Phase-WIDTH inputs, without invert bit and with state-controlled pulse forming network

One may think, that ON-OFF surpassing is of no issue, if, instead of inputting the phases for pulse ON and OFF, the phase for pulse ON and the phase difference corresponding to the pulse WIDTH are inputted, see Fig. 7. However, internally, the value for phase OFF must be recreated, by adding the WIDTH value to the ON value. Thereafter, a comparator, and two multiplexers are used again, as in Section V-B, to correct for the case ON trailing OFF. The remaining blocks are implemented the same as in Section V-A.

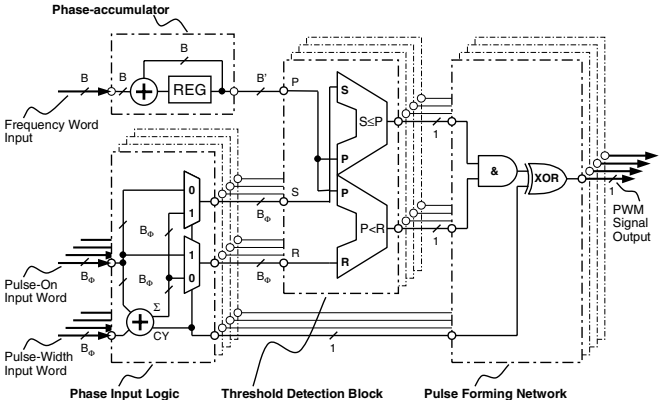

Fig. 7. PWM generator, with phase-ON and WIDTH inputs, without invert control signal.

D. Phase-ON and Phase-OFF inputs, without invert bit and with transition edge-controlled pulse forming network via a state machine

A synchronous, i.e. clocked by global clock, pulse forming network may be necessary, if PWM signals must be latched to an external clock, see Fig. 8. This is implemented by two

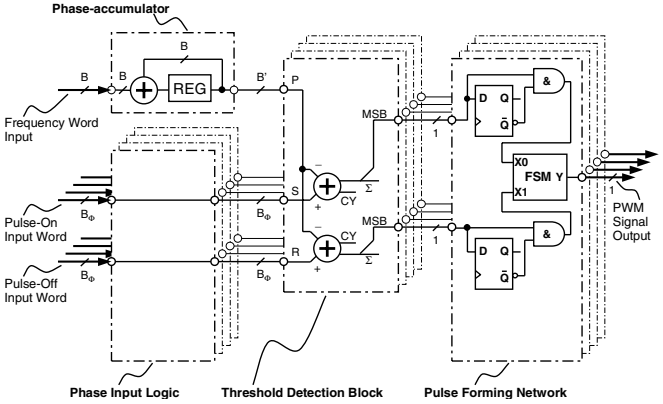

Fig. 8. PWM generator, with phase-ON and OFF inputs, with transition edge controlled via state machine (see Table II).

state-transition sensing D-flipflops, and a subsequent simple finite state machine. The state machine may be defined by the state sequence truth table defined in Table II, and may be implemented in hardware with a feedback structure by means of a 4:1 multiplexer or, directly, by two NORs and a D-flipflop. As inputs to the pulse forming network, the 'running'

TABLE II
CHARACTERIZATION OF FINITE STATE MACHINE , SEE FIG.8 & 9

X1	X0	Y_n	Y_{n+1}
0	0	Y_n	Y_n
0	1	-	1
1	0	-	0
1	1	-	0

phase P must be subtracted from each S and R, whereafter only the most significant bit (MSB), not the CARRY (CY) is then entered into the pulse forming network. Note, that this transition edge-controlled scheme does not require pulse inversion, as this function is incorporated here in the specific implementation for the threshold detection block and the pulse forming network. The state machine is constrained to eliminate the case ON trailing OFF.

E. Phase-ON and Phase-WIDTH inputs, without invert bit and with transition edge-controlled pulse forming network via a state machine

If, instead of inputting the phases for pulse ON and OFF, the phase for pulse ON and the phase difference corresponding to the pulse WIDTH are inputted, the case of Section V-D may be modified similar to the case of Section V-C, i.e. by adding the input of the phase difference representing the pulse WIDTH to the input value for pulse ON, although with the difference, that here, the specific state machine logic makes the invert-interchange function unnecessary, see Fig. 9, as the state machine is constrained to eliminate the case ON trailing OFF.

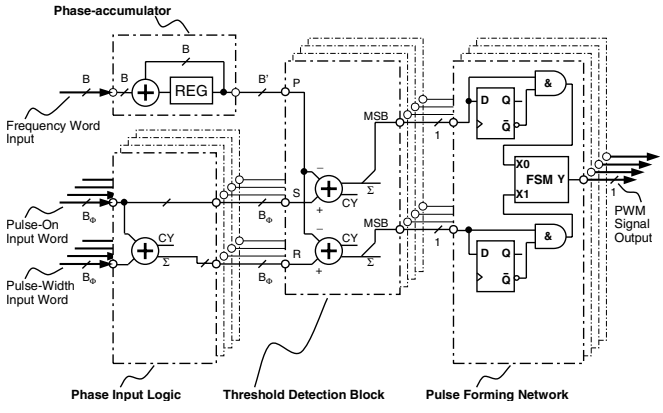

Fig. 9. PWM generator, with phase-ON and WIDTH inputs, with transition edge controlled via state machine (see Table II).

VI. SAMPLE HARDWARE IMPLEMENTATIONS

The fully digital design permits implementation into any semiconductor technology supporting a digital design flow. To test design and performance capabilities, three sample implementations were chosen, which are complementary in their scope: an ASIC in 0.35μm CMOS, a standard complex programmable logic device (CPLD) from ALTERA, the MAX7256AE for 3.3V operation, and a commercial test and development board 'BenOne' from Nallatech, populated with daughter board 'BenADDA', containing the XILINX Virtex II XC2V2000FG676A field programmable gate array (FPGA). These three sample implementations differ in their performance characteristics.

A. ASIC implementation

The architecture described in Section V-D was implemented as a subdesign on a larger multipurpose test chip fabricated in 0.35μm, 3-metal layer, CMOS technology from AMS Austria Mikrosysteme via Europractice service in a multi wafer run. Although the design was optimized with moderate area constraints and relaxed time constraints, a clock speed performance of about 170 MHz was reached. The critical delay path was located in the 32-bit phase-accumulator. In the test chip, as clock source can be used both an external clock or an internally generated clock by a ring oscillator, properly divided to meet design performance. The summary of the actual characteristic performance parameters are shown in Table III.

TABLE III
ASIC IMPLEMENTATION IN 0.35μm CMOS

Symbol	Function	Interface/control	Test chip parameters	
f_C	System clock	External/Int. Ring-Osc.	max. 170 Mhz	
f_{PWM}	Frequency	B (31 bit/int. 32 bit)	max. 85 MHz, min. 0 Hz res. 0.04Hz@ f_C =170Mhz res. 0.02Hz@ f_C =100Mhz res. 0.00002Hz@ f_C =40khz	
R	Rising edge phase	B_Φ (8 bit)	1.4°	97.7ns @ f_{PWM}=40kHz
S	Falling edge phase	B_Φ (8 bit)	1.4°	

B. CPLD/FPGA implementation

The CPLD design was developed for an actual industrial application to generate drive signals for a piezo electric motor. While this application requires only moderate PWM frequencies up to about 100 kHz and more characteristically about 40 kHz, the precision requirements to set phase and frequency are very high. The reason for this is, that the motor operates close to a resonance condition for surface waves within the piezo crystal, and phase control sets the rotary speed (revolutions). The higher the phase precision, the lower speed may be realized.

The design was optimized for the specific gate and logic cell capacity of the ALTERA MAX 7256 AE, and imbedded in a versatile development board, containing a JTAG interface operating under the IEEE 1149.1 standard to program the CPLD. Signal parameter input into the embedded PWM

generator takes place through a generic parallel interface, in this specific case connected on-board to a commercial piggy-back USB interface. This interface is socketed and may be replaced by any other standard interface components.

To implement the CPLD design, a flexible development environment was implemented first by means of the Nallatech board. For ease of use a Java based software package was developed featuring an easy-to-use graphical user interface (GUI) to control PWM parameters.

The actual characteristic performance parameters for both implementations are shown in Table IV.

TABLE IV
IMPLEMENTATION IN CPLD ALTERA AND FPGA XILINX DEVICE

Symbol	Function	Interface/controll	CPLD/FPGA parameters	
f_C	System clock	n.a.	max. 80 MHz – CPLD max. 120 Mhz – FPGA	
f_{PWM}	Frequency	B (31 bit/int. 32 bit)	max. 40 MHz, min. 0 Hz – CPLD max. 60 MHz, min. 0 Hz – FPGA res. 0.03Hz@f_C=120Mhz res. 0.02Hz@f_C=80Mhz res. 0.01Hz@f_C=40Mhz	
R	Rising edge phase	B_Φ (10 bit)	0.4°	12.5ns@f_{PWM}=80MHz
S	Falling edge phase	B_Φ (10 bit)	0.4°	

VII. CONCLUSION

A flexible block scheme for a novel PWM architecture operating in frequency domain was presented. This scheme offers a linear input word to frequency characteristic, and independent frequency and phase inputs, making it especially suitable for implementations fully in hardware. The fully digital architecture is readily cascaded for multi-channel applications with independent channel-to-channel pulse phases and duty cycles. It is available in high-level design language hardware description, and thus amenable to transfer to virtual any digital semiconductor technology. Test designs were implemented as ASIC in 0.35μm CMOS, as standard off-the shelf FPGA, and as standard off-the shelf CPLD.

REFERENCES

[1] M. Barr, "Pulse Width Modulation," *Embedded Systems Programming*, vol. 14, no. 10, pp. 103–104, Sept. 2001.

[2] M. Nagata, J. Funakoshi, and A. Iwata, "A PWM signal processing core circuit based on a switched current integration technique," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 1, pp. 53–60, Jan. 1998.

[3] H.-K. Yang and E. I. El-Masry, "A CMOS current-mode PWM technique for analog neural network implementations," in *Proc. of the IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 6, May 30 – June 2 1994, pp. 355–358.

[4] A. M. Trzynadlowski, "An overview of modern PWM techniques for three-phase, voltage-controlled, voltage-source inverters," in *Proc. of the IEEE International Symposium on Industrial Electronics (ISIE)*, vol. 1, June 17–20 1996, pp. 25–39, invited paper.

[5] Z. Yu, A. Mohammed, and I. Panahi, "A review of three PWM techniques," in *Proc. of the American Control Conference*, vol. 1, Albuquerque, NM, USA, June 04–06 1997, pp. 257–261.

[6] V. Oleschuk and F. Blaabjerg, "Direct synchronized PWM techniques with linear control functions for adjustable speed drives," in *Proc. of the 17th Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, vol. 1, Mar. 10–14 2002, pp. 76–82.

[7] A. M. Omar, N. A. Rahim, and S. Mekhilef, "Three-phase synchronous PWM for flyback converter with power-factor correction using FPGA ASIC design," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 1, pp. 96–106, Feb. 2004.

[8] A. M. Trzynadlowski, Z. Wang, J. Nagashima, C. Stancu, and M. Zelechowski, "Comparative investigation of PWM techniques for a new drive for electric vehicles," *IEEE Transactions on Industry Applications*, vol. 39, no. 5, pp. 1396–1403, Sept. – Oct. 2003.

[9] ——, "Comparative investigation of PWM techniques for General Motors' new drive for electric vehicles," in *Record of the Industry Applications Conference. 37th IAS Annual Meeting*, vol. 3, Oct. 13–18 2002, pp. 2010–2015.

[10] A. Richter, P. Rydlo, and M. Kolář, "Ultrasonic piezoceramics motor driven by PWM modulation with higher harmonic suppression," in *Proc. of the 14th International Conference on Electrical Drives and Power Electronics (EDPE)*, High Tatras, Slovak Republic, Sept. 24–26 2003, pp. 244–247, the proceedings are available in printed form (ISBN 80-89061-77-X) and in electronics form on the CD-ROM.

[11] R. L. Kirlin, M. M. Bech, and A. M. Trzynadlowski, "Analysis of power and power spectral density in PWM inverters with randomized switching frequency," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 2, pp. 486–499, Apr. 2002.

[12] S. Berto, S. Bolognani, M. Ceschia, A. Paccagnella, and M. Zigliotto, "FPGA-based random PWM with real-time dead time compensation," in *Proc. of the IEEE 34th Annual Power Electronics Specialist Conference (PESC)*, vol. 2, June 15–19 2003, pp. 513–518.

[13] K. Schade, I. Janiszewski, H. Meuth, H.-P. Schöner, and M. Schreiner, "A versatile multi-channel microelectronic digital PWM signal generator for piezo motor control," in *Proc. of the 9th International Conference on New Actuators and the 3rd International Exhibition on Smart Actuators and Drive Systems*, Bremen, Germany, June 14–16 2004, pp. 483–486, poster presentation.

[14] V. Oleschuk and F. Blaabjerg, "Synchronized scheme of continuous space-vector PWM with the real-time control algorithms," in *Proc of the IEEE 35th Annual Power Electronics Specialists Conference (PESC)*, vol. 2, June 20–25 2004, pp. 1207–1213.

[15] A. Kwasinski, P. T. Krein, and P. L. Chapman, "Time domain comparison of pulse-width modulation schemes," *IEEE Power Electronics Letters*, vol. 1, no. 3, pp. 64–68, Sept. 2003.

[16] H. Meuth, I. Janiszewski, and K. Schade, "Signalgenerator für pulsweitenmodulierte Signale auf rein digitaler Basis," Deutsches Patent- und Markenamt, 80297 München, Gebrauchsmuster/Registered design 20 2004 011 273.6, Jan. 20 2005, Tag der Eintragung/Enrollment 17.07.2004. A patent application is pending.